



Java OOP Basics Exam Preparation – Paw Inc.



Paw Incorporative is a large corporation that has the grand mission to rescue all street dogs and cats, shelter them, and possibly find a family to adopt them. You have been tasked to create a software program that oversees the operations of Paw Inc.

Overview

Paw Inc. has two main types of centers – Cleansing centers and Adoption centers.

The Cleansing centers' main function is to cleanse the animals of all types of dirt, diseases, or any parasites on them, so that they can be suitable for adoption.

The Adoption centers' main function is to provide the functionality of adopting an **already cleansed** animal.

The animals are first registered in the Adoption centers. However, they can **ONLY** be adopted if they are first cleansed.

Upon receiving the command, a particular Adoption center sends all of its stored animals to a particular Cleansing center.

Upon receiving the command, a particular Cleansing center cleanses all of its stored animals, and returns each stored animal to the Adoption center it received it from.

Paw Inc. works with two types of animals – dogs and cats.

Dogs are not quite smart, but they are loyal, so they have a particular amount of commands they have learned. Cats are quite intelligent, so they have an estimated intelligence coefficient.

Task 1: Classes and Data Encapsulation - 40 points

You need to implement the class structure for the components of Paw Inc.

The Centers:

- **Center** – stores common features of all centers.
- **CleansingCenter** – has a **name** and **stored animals**. Provides functionality for cleansing animals.
 - The name will be a string, which could consist of **any ASCII character**.
 - The stored animals can be many and can be both **dogs** and **cats**.
 - Upon receiving the cleansing command, it cleanses **all of the stored animals** it has, and **returns each** to the **Adoption center it received it from**.
- **AdoptionCenter** – has a name and stored animals. Provides functionality for sending animals for cleansing and adopting animals.
 - The name will be a string, which could consist of **any ASCII character**.
 - The stored animals can be many and can be both **dogs** and **cats**.

- Upon receiving the send for cleanse command, it sends **all of its stored animals** to the specified **Cleansing center**, so that they can be cleansed.
- Upon receiving the adoption command, **all of the cleansed animals, currently** in the **adoption center**, get **adopted** and **removed** from the stored animals.

The Animals:

- **Animal** – Stores common features for all animals.
 - Provide a **mandatory** constructor (**String, int**)
- **Dog** – has a **name, age, cleansing status** and **amount of commands** it has learned.
 - The name of the dog is a string, which can consist of **any ASCII character**.
 - The age of the dog is an **integer**.
 - The amount of commands of the dog is an **integer**.
- **Cat** – has a **name, age, cleansing status** and **intelligence coefficient**.
 - The name of the cat is a string, which can consist of **any ASCII character**.
 - The age of the cat is an **integer**.
 - The intelligence coefficient of the cat is an **integer**.

By default, all animals are **UNCLEANSED** upon registration.

Make sure no data is revealed completely, unless it is absolutely needed for it to be. Make sure you hide all data and also make sure that the input data follows the rules, specified in **Task 1**.

Task 2: Hierarchy - 20 points

You need to implement proper hierarchy for the provided classes.

Task 3: Application Logic - 20 points

The functionality of Paw Inc. is interpreted through commands. You need a class that handles commands.

AnimalCenterManager – stores information about all animal centers and handles commands.

It should have the following functionality:

- **void registerCleansingCenter(String name)**
- **void registerAdoptionCenter(String name)**
- **void registerDog(String name, int age, int learnedCommands, String adoptionCenterName)**
- **void registerCat(String name, int age, int intelligenceCoefficient, String adoptionCenterName)**
- **void sendForCleansing(String adoptionCenterName, String cleansingCenterName)**
- **void cleanse(String cleansingCenterName)**
- **void adopt(String adoptionCenterName)**
- **void printStatistics()**

There are several commands you need to have in mind.

Commands

- **RegisterCleansingCenter | {name}**
 - Registers a **Cleansing center** with the **given name**.
- **RegisterAdoptionCenter | {name}**
 - Registers an **Adoption center** with the **given name**.
- **RegisterDog | {name} | {age} | {learnedCommands} | {adoptionCenterName}**
 - Registers a dog with the **given name**, **age** and **amount of learned commands** in the **given Adoption center**.
- **RegisterCat | {name} | {age} | {intelligenceCoefficient} | {adoptionCenterName}**
 - Registers a cat with the **given name**, **age** and **intelligence coefficient** in the **given Adoption center**.
- **SendForCleansing | {adoptionCenterName} | {cleansingCenterName}**
 - Sends **all UNCLEANSED** animals from the **given Adoption center** to the **given Cleansing center**.
- **Cleanse | {cleansingCenterName}**
 - Cleanses **all animals** from the **given Cleansing center** and **returns** them to their **corresponding Adoption centers**.
- **Adopt | {adoptionCenterName}**
 - **All CLEANSED** animals from the **given Adoption center** get **adopted**, and **removed** from the stored animals.
- **Paw Paw Pawah**
 - This command **terminates** the input sequence.

You can safely assume that there will **be NO invalid** input commands. There will **be NO duplicate names** for the Cleansing and Adoption centers. There will **be NO nonexistent** centers in the commands.

Task 4: Input / Output - 20 points

No output should be printed as the **result of executing commands**.

The only output you should print is when you receive the input sequence – terminating command. Then you should print the following:

“Paw Incorporative Regular Statistics

Adoption Centers: {amountOfAdoptionCenters}

Cleansing Centers: {amountOfCleansingCenters}

Adopted Animals: {adoptedAnimal1Name}, {adoptedAnimal2Name}...

Cleansed Animals: {cleansedAnimal1Name}, {cleansedAnimal2Name}...

Animals Awaiting Adoption: {amountOfAnimalsWaitingForAdoption}

Animals Awaiting Cleansing: {amountOfAnimalsWaitingForCleansing}”

You need to present the count of all registered Adoption and Cleansing centers.

You need to show all animals, that were adopted, in **alphabetical order**.

You need to show all animals, that were cleansed, in **alphabetical order**.

The comparison is **case-sensitive**.

In case of no such animals, just print **“None”**.

You need to show the amount of all **CLEANSED** animals, waiting to be adopted in the **Adoption centers**.

You need to show the amount of all **UNCLEANSED** animals, waiting to be cleansed in the **Cleansing centers**.

Note: For the uncleaned animals, you need to print the count of only those, that are in the **Cleansing centers**.

Input

- The input will come in the form of commands, in the format specified above.
- The input sequence ends when the command “Paw Paw Pawah” is received.

Output

- The **only** output you need to print is **the final one**, when the input sequence ends.

Constraints

- The names of the centers and the animals can consist of any ASCII characters.
- The input age of the animals will be a valid integer in range $[0, 2^{31} - 1]$.
- The input amount of learned commands for the dogs will be a valid integer in range $[0, 2^{31} - 1]$.
- The input intelligence coefficient for the cats will be a valid integer in range $[0, 2^{31} - 1]$.
- You can safely assume that all commands will follow the format specified above.
- You can safely assume there will be **NO duplicate names** for the centers or the animals.
- Allowed time/memory: 250ms/16MB.

Examples

Input	Output
RegisterAdoptionCenter Destiny RegisterCleansingCenter Hope RegisterDog Sharo 5 20 Destiny RegisterCat Argent 1 200 Destiny SendForCleansing Destiny Hope Cleanse Hope Adopt Destiny RegisterAdoptionCenter Wisdom RegisterDog Rex 12 100 Wisdom RegisterDog Bolt 2 100 Wisdom SendForCleansing Wisdom Hope Paw Paw Pawah	Paw Incorporative Regular Statistics Adoption Centers: 2 Cleansing Centers: 1 Adopted Animals: Argent, Sharo Cleansed Animals: Argent, Sharo Animals Awaiting Adoption: 0 Animals Awaiting Cleansing: 2

Input	Output
RegisterAdoptionCenter Rebellion RegisterAdoptionCenter Sentinel RegisterCleansingCenter Sanctuary RegisterCleansingCenter Liberty RegisterCat Mr.Whiskas 1 1 Sentinel RegisterCat Gosho 20 100 Rebellion RegisterCat Chukky 10 10 Rebellion Adopt Sentinel RegisterDog Sirius 5 1000 Rebellion SendForCleansing Sentinel Sanctuary SendForCleansing Rebellion Sanctuary Cleanse Sanctuary Adopt Rebellion	Paw Incorporative Regular Statistics Adoption Centers: 2 Cleansing Centers: 2 Adopted Animals: Chukky, Gosho, Sirius Cleansed Animals: Chukky, Gosho, Mr.Whiskas, Sirius Animals Awaiting Adoption: 1 Animals Awaiting Cleansing: 1

RegisterDog Husku 1 20 Sentinel SendForCleansing Sentinel Liberty RegisterCat Nero 1 120 Sentinel Paw Paw Pawah	
--	--

Bonus Task: Castration Center - 20 points

The bonus task includes the implementation of a third type of centers, called – Castration centers.

Structure

- **Castration Center** – has a **name** and **stored animals**. Provides functionality for castrating animals.
 - The name will be a **string**, which could consist of **any ASCII character**.
 - The stored animals can be many and can be both **dogs** and **cats**.
 - Upon receiving the castration command, it castrates **all of the stored animals it has**, and returns each **to the Adoption center it received it from**.

Functionality

- **RegisterCastrationCenter | {name}**
 - Registers a **Castration center** with the **given name**.
- **SendForCastration | {adoptionCenterName} | {castrationCenterName}**
 - Sends **all unclesed** animals from the **given Adoption center** to the **given Castration center**.
- **Castrate | {castrationCenterName}**
 - Castrates **all animals** from the **given Castration center** and **returns** them to their **corresponding Adoption centers**.
- **CastrationStatistics**
 - Prints statistics about all the animals that have been castrated.
 - The format is:
“Paw Inc. Regular Castration Statistics
Castration Centers: {amountOfCastrationCenters}
Castrated Animals: {castratedAnimal1Name}, {castratedAnimal2Name}...”

The animals must be presented in **alphabetical order**. The comparison is **case-sensitive**.

In case of no such animals, just print **“None”**.

Bonus Example

Input	Output
RegisterAdoptionCenter Destiny RegisterCleansingCenter Hope RegisterCastrationCenter Razor RegisterCat Stacy 1 100 Destiny RegisterCat Cindy 1 110 Destiny SendForCastration Destiny Razor Castrate Razor CastrationStatistics SendForCleansing Destiny Hope Paw Paw Pawah	Paw Inc. Regular Castration Statistics Castration Centers: 1 Castrated Animals: Sindy, Stacy Paw Incorporative Regular Statistics Adoption Centers: 1 Cleansing Centers: 1 Adopted Animals: None Cleansed Animals: None Animals Awaiting Adoption: 0 Animals Awaiting Cleansing: 2